

13、系统睡眠唤醒--定时器唤醒

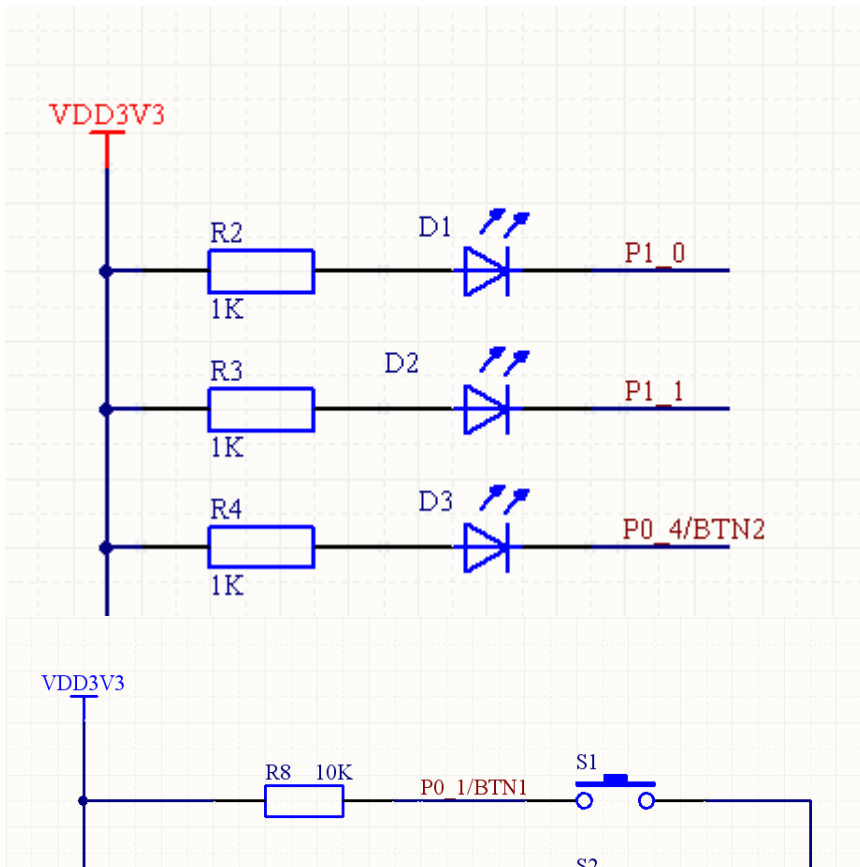
1. 实验目的

- 1) 为什么要睡眠? Zigbee 的特点就是远距离低功耗的无线传输设备, 节点模块闲时可以进入睡眠模式, 在需要传输数据时候进行唤醒, 能进一步节省电量。
- 2) 掌握几种系统电源模式的基本设置及切换。系统电源有以下几种管理模式: 全功能模式, 高频晶振 (16M 或者 32M) 和低频晶振 (32.768K RCOSC/XOSC) 全部工作, 数字处理模块正常工作。
PM1: 高频晶振 (16M 或者 32M) 关闭, 低频晶振 (32.768K RCOSC/XOSC) 工作, 数字核心模块正常工作。
PM2: 低频晶振 (32.768K RCOSC/XOSC) 工作, 数字核心模块关闭, 系统通过 RESET, 外部中断或者睡眠计数器溢出唤醒。
PM3: 晶振全部关闭, 数字处理核心模块关闭, 系统只能通过 RESET 或外部中断唤醒。此模式下功耗最低。
- 3) 将睡眠模式下的 CC2530 通过定时器唤醒, 观察 LED 闪烁现象

2. 实验设备

硬件: PC 机一台 ZB 网关 (底板、核心板、仿真器、USB 线) 一套
软件: 2000/XP/win7 系统, IAR 8.10 集成开发环境

3. 实验相关电路图



由于发光二极管单向导电特性，即只有在正向电压（二极管的正极接正，负极接负）下才能导通发光。P1.0 引脚接发光二极管(D1)的负极,所以 P1.0 引脚输出低电平 D1 亮，P1.0 引脚输出高电平 D1 熄灭。

4．实验分析及相关寄存器

相关寄存器 PCON，SLEEPCMD, ST0，ST1，ST2，如下表所示：（CC2530 中文数据手册完整版.pdf）

寄存器	作用	描述
PCON (0x87)	供电模式控制	Bit[0] 供电模式控制。写 1 到该位强制设备进入 SLEEP. MODE (注意 MODE=0x00 且 IDLE = 1 将停止 CPU 内核活动) 设置的供电模式, 这位读出来一直是 0。当活动时, 所有的使能中断将清除这个位, 设备将重新进入主动模式。
SLEEP_CMD (0xBE)	睡眠模式控制	Bit[1:0] 供电模式设置 00 : 主动/空闲模式 01 : 供电模式 1 10 : 供电模式 2 11 : 供电模式 3
ST0		睡眠计数器数据 Bit[7:0]
ST1		睡眠计数器数据 Bit[15:8]
ST2		睡眠计数器数据 Bit[23:16]

设置睡眠时间具体配置如下:

```
sleepTimer |= ST0;
sleepTimer |= (ulong)ST1 << 8;
sleepTimer |= (ulong)ST2 << 16;
sleepTimer += ((ulong)sec * (ulong)32768);
ST2 = (uchar)(sleepTimer >> 16);
ST1 = (uchar)(sleepTimer >> 8);
ST0 = (uchar) sleepTimer;
```

配置完毕后 sleepTimer 与 ST2<<16|ST1<<8|ST0 相差即为睡眠秒数

5. 源码分析

```

/*****
* 文件名: main.c
* 描述: 设置定时器让系统在设定的时间被唤醒,每次唤醒 LED1 闪烁 3 下提示用户
*****/

#include <ioCC2530.h>

```



```
typedef unsigned char uchar;  
typedef unsigned int uint;  
typedef unsigned long ulong;
```

```
#define LED1 P1_0      //P1.0 口控制 LED1  
#define LED2 P1_1      //P1.1 口控制 LED2
```

```
/******
```

```
* 名 称: DelayMS()  
* 功 能: 以毫秒为单位延时 16M 时约为 535,系统时钟不修改默认为 16M  
* 入口参数: msec 延时参数, 值越大, 延时越久  
* 出口参数: 无
```

```
***** /
```

```
void DelayMS(uint msec)
```

```
{  
    uint i,j;  
  
    for (i=0; i<msec; i++)  
        for (j=0; j<535; j++)  
}
```

```
/******
```

```
* 名 称: InitLed()  
* 功 能: 设置 LED 灯相应的 IO 口  
* 入口参数: 无  
* 出口参数: 无
```

```
***** /
```

```
void InitLed(void)
```

```
{  
    P1DIR |= 0x03;      //P1.0 定义为输出口  
    LED1 = 1;           //LED1 灯上电默认为熄灭  
}
```

```
/******
```

```
* 名 称: SysPowerMode()
```

```
* 功 能: 设置系统工作模式
```

```
* 入口参数: mode 等于 0 为 PM0 1 为 PM1 2 为 PM2 3 为 PM3
```

```
* 出口参数: 无
```

```
*****/
```

```
void SysPowerMode(uchar mode)
```

```
{  
    if(mode < 4)  
    {  
        SLEPCMD |= mode; //设置系统睡眠模式  
        PCON = 0x01;      //进入睡眠模式 ,通过中断唤醒  
    }  
    else  
        PCON = 0x00;      //通过中断唤醒系统  
}
```

```
/******
```

```
* 名 称: ST_ISR(void) 中断处理函数
```

```
* 描 述: #pragma vector = 中断向量 , 紧接着是中断处理程序
```

```
*****/
```

```
#pragma vector = ST_VECTOR
```

```
__interrupt void ST_ISR(void)
```

```
{  
    STIF = 0;      //清标志位  
    SysPowerMode(4); //进入正常工作模式  
}
```

```
/******
```

```
* 名 称: SysPowerMode()
```



* 功 能: 初始化休眠定时器,设定后经过指定时间自行唤醒

* 入口参数:

* 出口参数: 无

*****/

void InitSleepTimer(void)

```
{
    ST2 = 0X00;
    ST1 = 0X0F;
    ST0 = 0X0F;
    EA = 1;    //开中断
    STIE = 1; //睡眠定时器中断使能 0 : 中断禁止    1 : 中断使能
    STIF = 0; //睡眠定时器中断标志 0 : 无中断未决  1 : 中断未决
}
```

/*****

* 名 称: Set_ST_Period()

* 功 能: 设置睡眠时间

* 入口参数: sec 睡眠时间

* 出口参数: 无

*****/

void Set_ST_Period(uint sec)

```
{
    ulong sleepTimer = 0;

    sleepTimer |= ST0;
    sleepTimer |= (ulong)ST1 << 8;
    sleepTimer |= (ulong)ST2 << 16;
    sleepTimer += ((ulong)sec * (ulong)32768);
    ST2 = (uchar)(sleepTimer >> 16);
    ST1 = (uchar)(sleepTimer >> 8);
    ST0 = (uchar) sleepTimer;
}
```

```
}
```

```
/******
```

```
* 程序入口函数
```

```
*****/
```

```
void main(void)
```

```
{
```

```
    uchar i=0;
```

```
    InitLed();           //设置 LED 灯相应的 IO 口
```

```
    InitSleepTimer();    //初始化休眠定时器
```

```
    while(1)
```

```
    {
```

```
        for (i=0; i<6; i++) //LED1 闪烁 3 次提醒用户将进入睡眠模式
```

```
        {
```

```
            LED1 = ~LED1;
```

```
            DelayMS(500);
```

```
        }
```

```
        Set_ST_Period(5); //设置睡眠时间,睡眠 5 秒后唤醒系统
```

```
        SysPowerMode(2); //重新进入睡眠模式 PM2
```

```
        LED2 = ~LED2;
```

```
    }
```

```
}
```

6.实验现象

开机后 LED1 闪 3 次后进入睡眠，睡眠 5 秒后，自系统动醒来，LED2 亮。LED1 闪 3 次后再次进入睡眠，睡眠 5 秒后，系统自动醒来，LED2 灭。LED2 亮灭交替__